Information theory

Concept of information (through an example)

Information content of data streams, information rate

Entropy and information

"I wish I could be as calm as JB when it comes to making decisions."

Concept of information (through an example)

Intuitive concept:

"informare" (Lat.) : „to give form to the mind”, or to teach, instruct somebody

Thus: „We can only change our minds, when we recieve ***information***.”

Or:
„a type of input to an organism or designed device” : Ecology, sensory input
(Smell of food → movement of animal)

Or:
„information is any type of pattern that influences
the formation or transformation of other patterns.”
(RNA sequence → Protein structure)

“I wish I could be as calm as JB when it comes to making decisions.”

Transmitting information – information content

Event and information:
What happened?

„Information content" of events:

-It is light traffic this morning

-It will rain tomorrow.

-I have won the lottery!

How can we *encode* information?

Transmitting information – information **coding**

in general

Information source — Which event occured from a set of possibilities?

↓ **en**coding — **En**coding: We represent **events** with **NUMBERS**

Transmission channel

↓ **de**coding — **De**coding: We reconstruct **events** from **NUMBERS**

Information reciever
destination — (news)

Transmitting information – information **coding**

in general

Information **source**

encoding

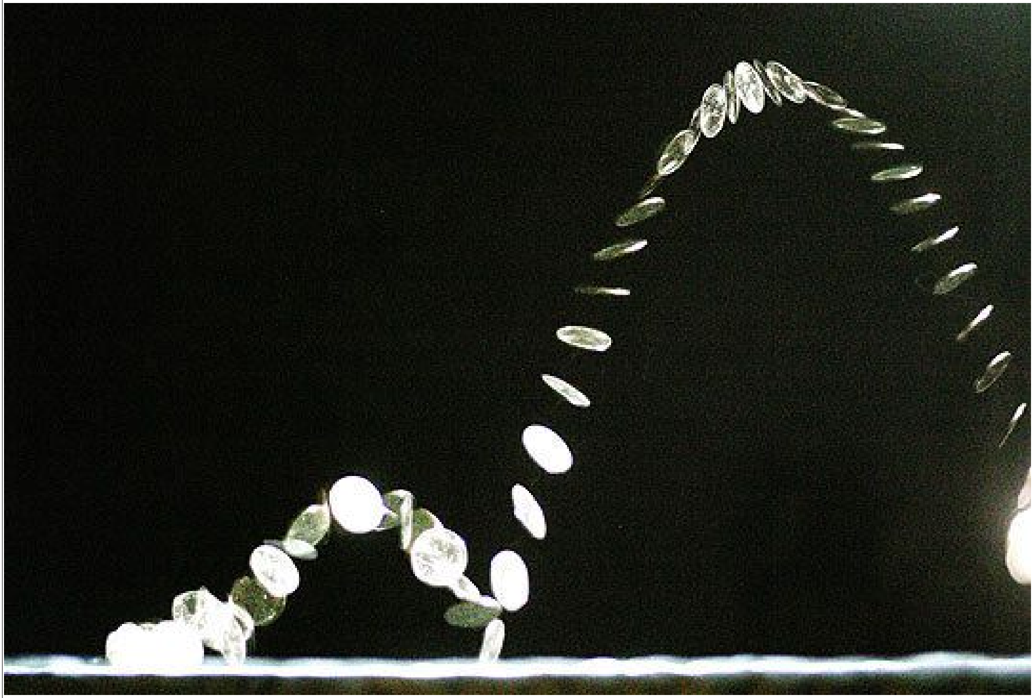
Transmission **channel**

decoding

Information reciever
destination

an example

Tossing a dime



Head or Tail?

Transmitting information – information **coding**

in general

Information **source**

encoding

Transmission **channel**

decoding

Information reciever
destination

an example

Which side is up?

encoding

Sides : Head or Tail
into Numbers: 1,0

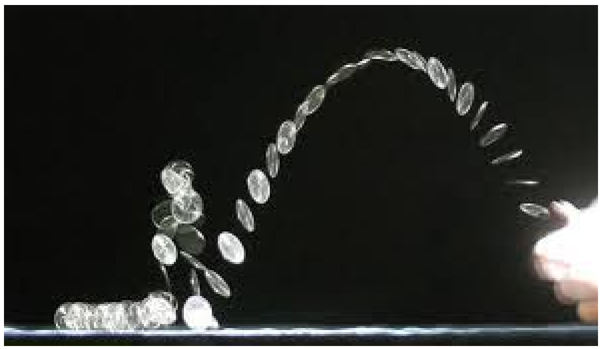
Speech, waves in the air, sms

decoding

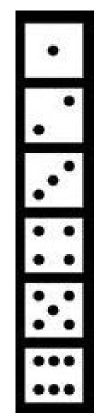
1,0 ⟶ head, tail

Decide who wins

Transmitting information – *digital* **coding**



Event		Number	Digital code
heads	:	1	1
tails	:	0	0

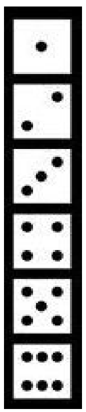


	:	1	001
	:	2	010
	:	3	011
	:	4	100
	:	5	101
	:	6	110

2-base numbers: example: $101_2 = 1*2^2+0*2^1+1*2^0 = 5_{10}$

bit = „binary digit”

Transmitting information – coding ***efficiency***



Event	Number	Digital code	Bits needed	Maximum number of events
:	1	001		
:	2	010		
:	3	011	**3**	**8**
:	4	100		
:	5	101		
:	6	110		
	7	111		
	0	000		

Here we only have 6 events,
but could encode 8 in 3 bits!

A better encoding:

$\{X_1X_2X_3\}$ group 3 events together : number of possibilities = $6^3 = 216$

$256 = 2^8$

Classic coding
3x3 bits = **9** bits → It is possible to encode 3 events in **8** bits

1 bit less!

Transmitting information – information content

Information content = how many bits do we *minimally* need to encode

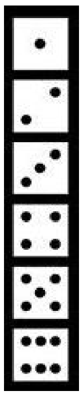
(This also gives the encoding efficiency limit)

How does this connect with intuitive information content?

	p	q	
-I have tossed a dime. Head or Tail?	½	½	No idea
-It is light traffic this morning	¼	¾	
-It will rain tomorrow.	1%	99%	
-I have won the lottery!	1/13,983,816	0.999....	Probably no win

Gained information is inverse proportional to the probability (p)

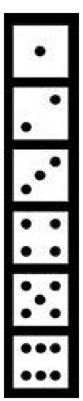
Transmitting information – measure of information



Fair	P_i	probability	code example	bits needed	p*(number of bits needed)
	1/6	0,17	000	3	0,5
	1/6	0,17	001	3	0,5
	1/6	0,17	010	3	0,5
	1/6	0,17	011	3	0,5
	1/6	0,17	100	3	0,5
	1/6	0,17	101	3	0,5

Expected number of bits needed: **3**

We can encode more efficiently here:

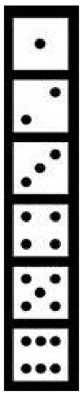


Loaded	P_i	probability	code example	bits needed	p*(number of bits needed)
	1/2	0,5	0	1	0,5
	1/4	0,25	10	2	0,5
	1/8	0,13	110	3	0,38
	1/16	0,06	1110	4	0,25
	1/32	0,03	11110	5	0,16
	1/32	0,03	11111	5	0,16

Expected number of bits needed: **1,94**

Transmitting information – measure of information

Fair



P_i	probability	code example	bits needed	p*(number of bits needed)
1/6	0,17	000	3	0,5
1/6	0,17	001	3	0,5
1/6	0,17	010	3	0,5
1/6	0,17	011	3	0,5
1/6	0,17	100	3	0,5
1/6	0,17	101	3	0,5

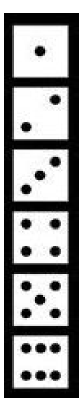
Here we do NOT Expect anything

Maximal uncertainity

Expected number of bits needed: **3**

Gained information is proportional to the number of bits needed

Loaded



P_i	probability	code example	bits needed	p*(number of bits needed)
1/2	0,5	0	1	0,5
1/4	0,25	10	2	0,5
1/8	0,13	110	3	0,38
1/16	0,06	1110	4	0,25
1/32	0,03	11110	5	0,16
1/32	0,03	11111	5	0,16

Here we *expect* „one” (most probable)

On average we **learn less**

Expected number of bits needed: **1,94**

Here the information content is less.

Transmitting information – measure of information

We have a way to define the information content ONLY with the probability, without the need of a specific encoding scheme

Shannon : define measure as: $$H = p \cdot \log_2\left(\frac{1}{p}\right)$$

It is also useful to calculate the information content *of a single event*:

$$I = \log_2\left(\frac{1}{p}\right)$$

Thus, the H=p*I is a weighted value of the information content, the weighting factor is the probability. This will be useful, if the **average** information content is needed.

$\log_2$: 2-base logarithm

Examples:

$\log_2(2) = 1$

$\log_2(4) = 2$

$\log_2(8) = 3$

Transmitting information – measure of information

Shannon $$H = p \cdot \log_2\left(\frac{1}{p}\right)$$ [bit]

If we have multiple events in the set, then it is a sum for every possible event:

$$H = \sum_i p_i \cdot \log_2\left(\frac{1}{p_i}\right) = \sum_i -p_i \cdot \log_2 p_i$$

other log-bases:
$\log_e$ (ln) : [nat]
$\log_{10}$ (lg) : [ban]

measure of information - entropy

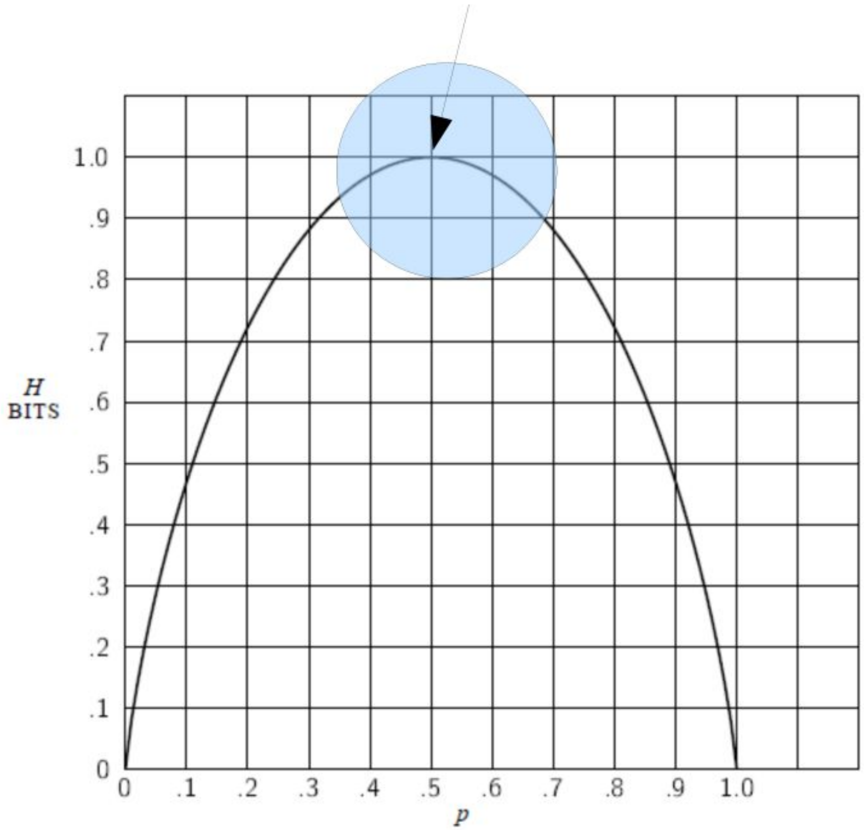
Fair dime: p = ½

no expectations
maximal uncertainity

Dime tossing

p

q = 1-p



$$H=\sum_i -p_i \cdot \log_2 p_i = -p \cdot \log_2 p - q \cdot \log_2 q = -p \cdot \log_2 p - (1-p) \cdot \log_2 (1-p)$$

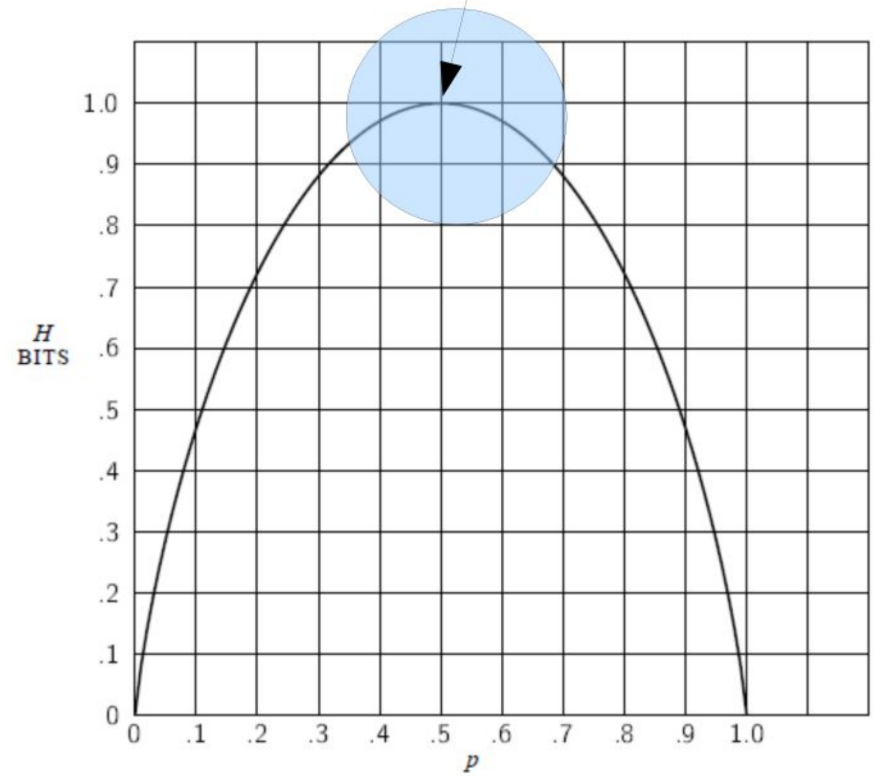
measure of information - entropy

Fair dime: $p = ½$

no expectations
maximal uncertainity

Dime tossing

p   q = 1-p



H has another name: **Shannon-entropy**

H has a maximum when we know nothing in advance (all p_i-s are equal, $p_i = 1/n$)

Expected outcomes are maximized: each state is equally probable

⇕

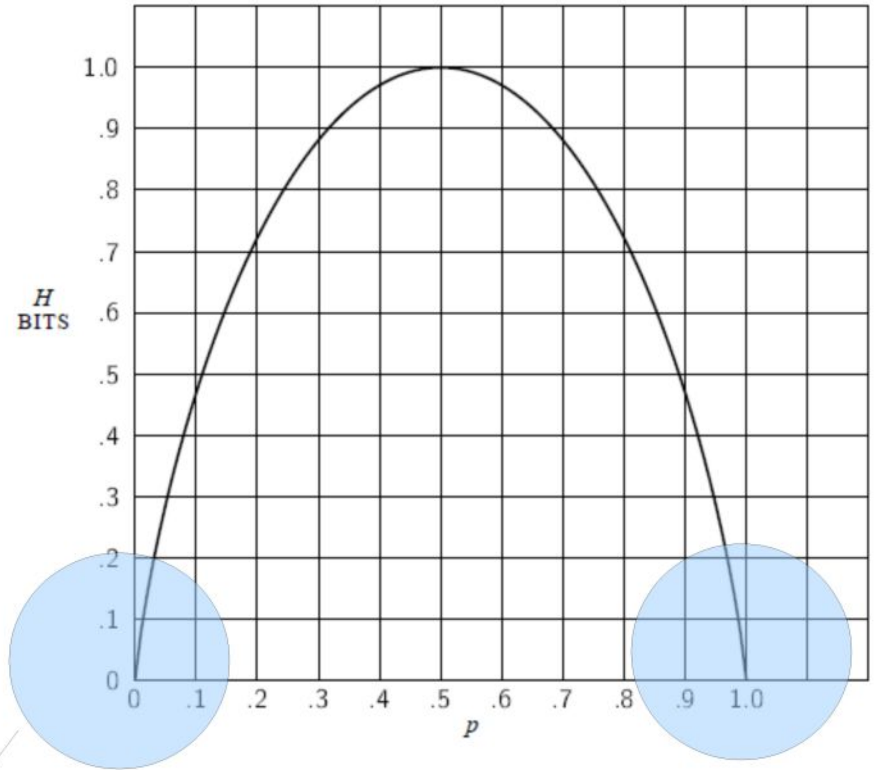
Physical entropy (S) has a maximum if the number of microstates is maximal.

measure of information - entropy

Dime tossing

p

q = 1-p



H has another name: **Shannon-entropy**

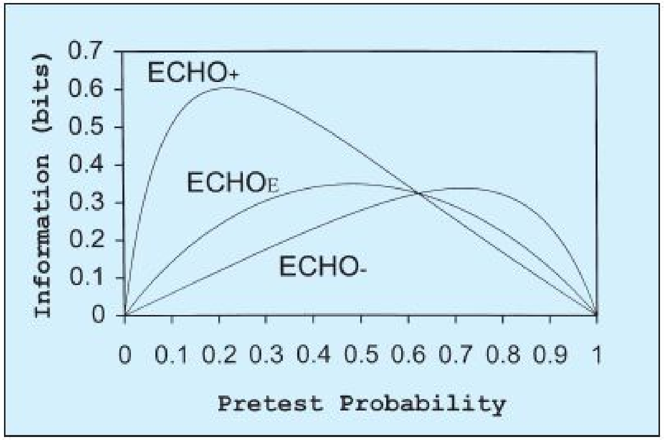
H vanishes ONLY if we are absolutely certain of the outcome: p=0 or p=1

⇕

Physical entropy (S) vanishes ONLY if there is exactly 1 microstate

Extension Material

Examples of usage in medicine



Bayes-theorem based methods:

The amount of information gained by performing a diagnostic test can be quantified by calculating the relative entropy between the posttest and pretest probability distributions

Application:
- Diagnostic tests
- expert systems

a_i: pretest probability

b_i: post test probability

$$D(b\|a) = \sum_{i=1}^{n} b_i \log_2(b_i/a_i)$$

Testing Situation	Pretest Probability of Disease	Test Operating Characteristics: Sensitivity/Specificity	Test Result	Posttest Probability of Disease	Information Gained
Breast cancer screening with mammography	0.01	0.75/0.94	Positive Negative	0.11 0.003	0.25 bits 0.006 bits
Mammography given palpable breast mass	0.2	0.80/0.90	Positive Negative	0.67 0.05	0.74 bits 0.13 bits
Screening for HIV with antibody test	0.001	0.99/0.998	Positive Negative	0.33 0.00001	2.4 bits 0.001 bits
Presence of tonsillar exudate in diagnosing infection with group A streptococci	0.1	0.45/0.84	Positive Negative	0.24 0.07	0.11 bits 0.01 bits
Colon cancer screening by fecal occult blood testing	0.005	0.40/0.90	Positive Negative	0.02 0.003	0.02 bits 0.0005 bits

Databases

Databases store information:

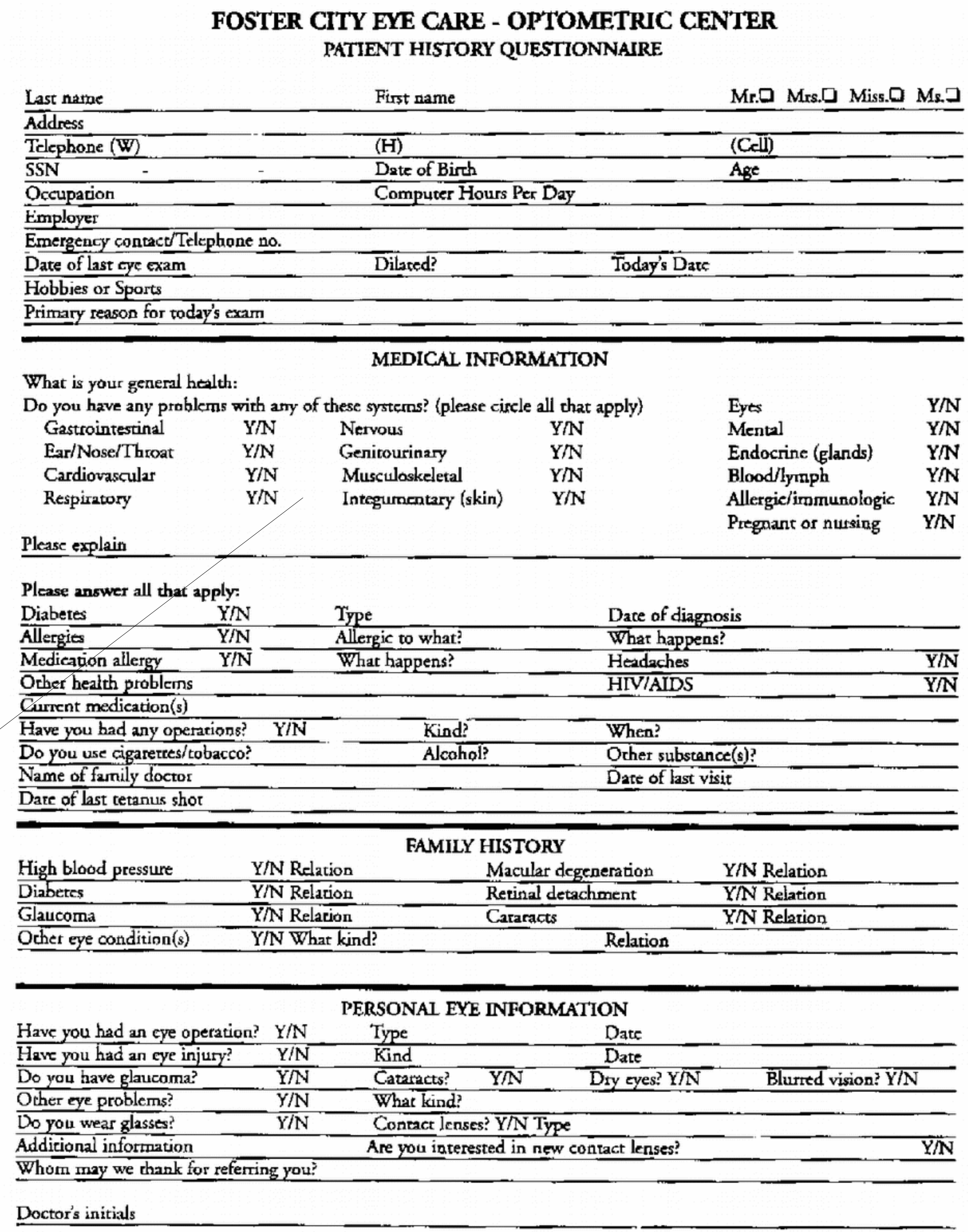
Databases are used for:
storage, structuring and extraction of ***information*** gathered previously.

Databases

Databases store information:

Databases are used for: storage, structuring and **extraction** of ***information*** gathered previously.

It is hard to ***extract*** or modify information stored on paper



FOSTER CITY EYE CARE - OPTOMETRIC CENTER

PATIENT HISTORY QUESTIONNAIRE

Last name First name Mr.❑ Mrs.❑ Miss.❑ Ms.❑

Address

Telephone (W) (H) (Cell)

SSN - - Date of Birth Age

Occupation Computer Hours Per Day

Employer

Emergency contact/Telephone no.

Date of last eye exam Dilated? Today's Date

Hobbies or Sports

Primary reason for today's exam

MEDICAL INFORMATION

What is your general health:

Do you have any problems with any of these systems? (please circle all that apply)

Gastrointestinal	Y/N	Nervous	Y/N	Eyes	Y/N
Ear/Nose/Throat	Y/N	Genitourinary	Y/N	Mental	Y/N
Cardiovascular	Y/N	Musculoskeletal	Y/N	Endocrine (glands)	Y/N
Respiratory	Y/N	Integumentary (skin)	Y/N	Blood/lymph	Y/N
				Allergic/immunologic	Y/N
				Pregnant or nursing	Y/N

Please explain

Please answer all that apply:

Diabetes Y/N Type Date of diagnosis

Allergies Y/N Allergic to what? What happens?

Medication allergy Y/N What happens? Headaches Y/N

Other health problems HIV/AIDS Y/N

Current medication(s)

Have you had any operations? Y/N Kind? When?

Do you use cigarettes/tobacco? Alcohol? Other substance(s)?

Name of family doctor Date of last visit

Date of last tetanus shot

FAMILY HISTORY

High blood pressure	Y/N Relation	Macular degeneration	Y/N Relation
Diabetes	Y/N Relation	Retinal detachment	Y/N Relation
Glaucoma	Y/N Relation	Cataracts	Y/N Relation
Other eye condition(s)	Y/N What kind?	Relation	

PERSONAL EYE INFORMATION

Have you had an eye operation? Y/N Type Date

Have you had an eye injury? Y/N Kind Date

Do you have glaucoma? Y/N Cataracts? Y/N Dry eyes? Y/N Blurred vision? Y/N

Other eye problems? Y/N What kind?

Do you wear glasses? Y/N Contact lenses? Y/N Type

Additional information Are you interested in new contact lenses? Y/N

Whom may we thank for referring you?

Doctor's initials

Databases – storing information

Name	Telephone	Insurance	Diagnosis	Physician	Room	Bed
Sam Small	(763) 865 345	Medicaid	Influenza	Dr. Barkins	37	2
Sara Goldmann	(691) 579 467	Medicare	Ascites	Dr. Magenheim	21	1
Dan Trideman	(691) 556 322	Medicaid	Malaria	Dr. Haydens	17	2
Bill Hardy	(691) 654 321	Medicare	Diabetes	Dr. Haydens	43	1
Bob Mindy	(691) 143 613	Tricare	Not yet known	Dr. Barkins	33	1
Yo Him	(691) 244 567	Medicare	Colitis	Dr. Magenheim	27	3
Kim Suhan	(691) 379 788	Tricare	Pneumonia	Dr. Barkins	37	1

Databases – storing information

Name	Telephone	Insurance	Diagnosis	Physician	Room	Bed
Sam Small	(763) 865 345	Medicaid	Influenza	Dr. Barkins	37	2
Sara Goldmann	(691) 579 467	Medicare	Ascites	Dr. Magenheim	21	1
Dan Trideman	(691) 556 322	Medicaid	Malaria	Dr. Haydens	17	2
Bill Hardy	(691) 654 321	Medicare	Diabetes	Dr. Haydens	43	1
Bob Mindy	(691) 143 613	Tricare	Not yet known	Dr. Barkins	33	1
Yo Him	(691) 244 567	Medicare	Colitis	Dr. Magenheim	27	3
Kim Suhan	(691) 379 788	Tricare	Pneumonia	Dr. Barkins	37	1

Table : ordered set of data (information)

Databases – storing information

	A	B	C	D	E	F	G
1	Name	Telephone	Insurance	Diagnosis	Physician	Room	Bed
2	Sam Small	(763) 865 345	Medicaid	Influenza	Dr. Barkins	37	2
3	Sara Goldmann	(691) 579 467	Medicare	Ascites	Dr. Magenheim	21	1
4	Dan Trideman	(691) 556 322	Medicaid	Malaria	Dr. Haydens	17	2
5	Bill Hardy	(691) 654 321	Medicare	Diabetes	Dr. Haydens	43	1
6	Bob Mindy	(691) 143 613	Tricare	Not yet known	Dr. Barkins	33	1
7	Yo Him	(691) 244 567	Medicare	Colitis	Dr. Magenheim	27	3
8	Kim Suhan	(691) 379 788	Tricare	Pneumonia	Dr. Barkins	37	1

Record : Information *grouped together (one ROW in a Table)*

Each row is a selected set of data

Every row has the same structure

Databases – storing information

Name	Telephone	Insurance	Diagnosis	Physician	Room	Bed
Sam Small	(763) 865 345	Medicaid	Influenza	Dr. Barkins	37	2
Sara Goldmann	(691) 579 467	Medicare	Ascites	Dr. Magenheim	21	1
Dan Trideman	(691) 556 322	Medicaid	Malaria	Dr. Haydens	17	2
Bill Hardy	(691) 654 321	Medicare	Diabetes	Dr. Haydens	43	1
Bob Mindy	(691) 143 613	Tricare	Not yet known	Dr. Barkins	33	1
Yo Him	(691) 244 567	Medicare	Colitis	Dr. Magenheim	27	3
Kim Suhan	(691) 379 788	Tricare	Pneumonia	Dr. Barkins	37	1

Column: data type

Databases – SQL

A relational database stores each key (information) ONCE, and it stores the connections between objects.

The database models the logic of the data set.

A Relational Model of Data for Large Shared Data Banks

E. F. Codd
IBM Research Laboratory, San Jose, California

Future users of large data banks must be protected from having to know how the data is organized in the machine (the internal representation). A prompting service which supplies such information is not a satisfactory solution. Activities of users at terminals and most application programs should remain unaffected when the internal representation of data is changed and even when some aspects of the external representation are changed. Changes in data representation will often be needed as a result of changes in query, update, and report traffic and natural growth in the types of stored information.

Existing noninferential, formatted data systems provide users with tree-structured files or slightly more general network models of the data. In Section 1, inadequacies of these models are discussed. A model based on *n*-ary relations, a normal form for data base relations, and the concept of a universal data sublanguage are introduced. In Section 2, certain operations on relations (other than logical inference) are discussed and applied to the problems of redundancy and consistency in the user's model.

KEY WORDS AND PHRASES: data bank, data base, data structure, data organization, hierarchies of data, networks of data, relations, derivability, redundancy, consistency, composition, join, retrieval language, predicate calculus, security, data integrity
CR CATEGORIES: 3.70, 3.73, 3.75, 4.20, 4.22, 4.29

1. Relational Model and Normal Form

1.1. Introduction

This paper is concerned with the application of elementary relation theory to systems which provide shared access to large banks of formatted data. Except for a paper by Childs [1], the principal application of relations to data systems has been to deductive question-answering systems. Levein and Maron [2] provide numerous references to work in this area.

In contrast, the problems treated here are those of *data independence*—the independence of application programs and terminal activities from growth in data types and changes in data representation—and certain kinds of *data inconsistency* which are expected to become troublesome even in nondeductive systems.

The relational view (or model) of data described in Section 1 appears to be superior in several respects to the graph or network model [3, 4] presently in vogue for non-inferential systems. It provides a means of describing data with its natural structure only—that is, without superimposing any additional structure for machine representation purposes. Accordingly, it provides a basis for a high level data language which will yield maximal independence between programs on the one hand and machine representation and organization of data on the other.

A further advantage of the relational view is that it forms a sound basis for treating derivability, redundancy, and consistency of relations—these are discussed in Section 2. The network model, on the other hand, has spawned a number of confusions, not the least of which is mistaking the derivation of connections for the derivation of relations (see remarks in Section 2 on the "connection trap").

Finally, the relational view permits a clearer evaluation of the scope and logical limitations of present formatted data systems, and also the relative merits (from a logical standpoint) of competing representations of data within a single system. Examples of this clearer perspective are cited in various parts of this paper. Implementations of systems to support the relational model are not discussed.

1.2. Data Dependencies in Present Systems

The provision of data description tables in recently developed information systems represents a major advance toward the goal of data independence [5, 6, 7]. Such tables facilitate changing certain characteristics of the data representation stored in a data bank. However, the variety of data representation characteristics which can be changed *without logically impairing some application programs* is still quite limited. Further, the model of data with which users interact is still cluttered with representational properties, particularly in regard to the representation of collections of data (as opposed to individual items). Three of the principal kinds of data dependencies which still need to be removed are: ordering dependence, indexing dependence, and access path dependence. In some systems these dependencies are not clearly separable from one another.

1.2.1. *Ordering Dependence.* Elements of data in a data bank may be stored in a variety of ways, some involving no concern for ordering, some permitting each element to participate in one ordering only, others permitting each element to participate in several orderings. Let us consider those existing systems which either require or permit data elements to be stored in at least one total ordering which is closely associated with the hardware-determined ordering of addresses. For example, the records of a file concerning parts might be stored in ascending order by part serial number. Such systems normally permit application programs to assume that the order of presentation of records from such a file is identical to (or is a subordering of) the